

REMARKS

The Examiner is thanked for the performance of a thorough search.

Prior to entry of this response, Claims 1-50 were pending in the application. By this response, no claims are added or canceled. Hence, Claims 1-50 are pending in the application upon entry of this response.

Claims 1, 20-39, 42-44, and 50 are amended herein.

SUMMARY OF THE REJECTIONS/OBJECTIONS

Claims 20-38 and 42-44 were rejected under 35 U.S.C. §101 as allegedly directed to non-statutory subject matter; and

Claims 1-50 were rejected under 35 U.S.C. § 102(e) as being anticipated by Fernandez et al., U.S. Patent No. 6,785,673 (hereinafter "*Fernandez*").

THE REJECTIONS NOT BASED ON THE PRIOR ART

Rejection under 35 U.S.C. §101

Claims 20-38 and 42-44 were rejected under 35 U.S.C. §101 as allegedly directed to non-statutory subject matter. Specifically, the Office Action asserts that because the specification describes computer-readable media as including transmission media, these computer-readable medium claims are considered as not falling within any of the four statutory categories of patentable subject matter recited in 35 U.S.C. §101.

Claims 20-38 and 42-44 are amended herein to substitute “computer-readable storage medium” for “computer-readable medium” to clearly place these claims in the statutory category of “a manufacture.” Hence, this rejection is overcome and is now moot. Reconsideration and withdrawal of the rejection of Claims 20-38 and 42-44 under 35 U.S.C. §101 is requested.

THE REJECTIONS BASED ON THE PRIOR ART

Rejection Under 35 U.S.C. § 102(e)

Claims 1-50 were rejected under 35 U.S.C. § 102(e) as allegedly anticipated by *Fernandez*.

Claim 1 recites the following:

detecting that a portion of a query execution plan to service a request for data will cause a first producer execution unit that will perform said portion, according to said query execution plan, to generate XML data for use by a second consumer execution unit in performing another portion of said query execution plan;

generating information to send to said first execution unit to cause said first execution unit to perform said portion of said query execution plan;

wherein said information would cause said first execution unit to generate said XML data in a first form that cannot be used by said second execution unit; and

annotating said information with an annotation that causes XML data generated by said first execution unit to be transformed to a canonical form for use by said second execution unit in performing said another portion of said query execution plan.

The Office Action relies on *Fernandez* for alleged teachings of each of the limitations recited in Claim 1. However, the disclosure of *Fernandez* is directed to quite a different context and approach than the subject matter of Claim 1. Therefore, the limitations of Claim 1 are not sufficiently taught or suggested in the *Fernandez* reference to support a valid anticipation rejection.

A query optimizer is a component of a database server that generates an execution plan to execute queries received by the database server. A query execution plan defines the steps

and operations performed by a database server to process a query. For example, the optimizer may examine possible access paths (e.g., index scan, sequential scan) and join algorithms (e.g. sort-merge join, hash join, nested loops). Thus, for example, a query execution plan may specify in detail what database tables need to be accessed and how, what tables need to be joined and how, whether or not data needs to be sorted and in what order, how portions of the plan may be divided up among multiple processes (e.g., in a parallel query execution plan), and the like.

In Claim 1, a query execution plan (or simply “query plan”) involves a producer execution unit (“producer”), by performing a portion of the query plan, generating XML data for a consumer execution unit (“consumer”) to use as input in performing another portion of the query plan. It is from the query plan that it is detected that the particular producer, in performing its portion of the query plan, will generate XML data for use by the particular consumer. Furthermore, once the XML data generated by the producer is transformed to the canonical form, the consumer is able to perform its portion of the query execution plan, e.g., using either the canonical form of XML data or a transformed form of the canonically-formed XML data which the consumer can use.

Thus, a **query execution plan** is clearly used in Claim 1. By contrast, *Fernandez* does not teach any use of a query execution plan. The Office Action points to FIGS. 6 and 7 for an alleged teaching of the first limitation of Claim 1, which includes use of a query execution plan. However, the “query planner” of *Fernandez* “partitions a view tree into one or more subtrees” from which corresponding SQL queries are generated (col. 38, lines 17-18), and from which it is determined whether the cost of two separate queries is more or less than the cost of a single query combining the two separate queries (col. 48, lines 21-32). Subtrees which are partitioned from a view tree is not the same as a query execution plan, because these subtrees do not define

the steps and operations performed by a database server to process a query, as does a query execution plan. Stated otherwise, a query execution plan goes beyond simply generating queries in that a query execution plan details precisely *how* to execute one or more queries relative to the manner in which corresponding data is stored, indexed, etc. Hence, *Fernandez* does not teach use of a query execution plan for **detecting that the producer will, by performing a portion of the execution plan, generate XML data for use by the consumer in performing another portion of the execution plan.**

Additionally, *Fernandez* does not teach or suggest the **use of multiple execution units for servicing a common data request**, e.g., according to a parallel query (PQ) framework implemented in a database server or across a distributed database, where each execution unit is a different process or thread. FIG. 6 of *Fernandez* does not show, teach, or suggest multiple execution units which work to service a common request. For FIG. 6 to map to the subject matter of Claim 1, not only would a real query execution plan (as referred to and known in the database arts) be needed rather than a view tree partitioning and query relation integration scheme (as in FIG. 6), but **FIG. 6 would also need to show that one process or thread generates the XML document and that a second process or thread uses the XML document to further perform part of the execution plan to further service the data request.** FIG. 6 clearly does not show such a multi-execution unit framework or architecture, nor does the remainder of *Fernandez* disclose such a multi-execution unit context. Consequently, *Fernandez* does not teach detecting that the producer will generate XML data for use by the consumer in performing another portion of the execution plan. As the title illustrates, *Fernandez* is about converting relational data into XML, and is not about managing XML data among multiple execution units so that XML data from a particular producer, in a

form that would not normally be compatible to a particular consumer, is made compatible to the consumer via the annotation-based transformation of the XML data output by the producer.

A further consequence of *Fernandez* not disclosing a multi-execution unit context is that ***Fernandez* does not teach that, but for the annotation, the producer would generate XML data in a first form that cannot be used by the consumer.** The concept that but for the annotation, the producer would generate XML data in a form that is unusable by the consumer, was originally in Claim 2. For the rejection of Claim 2, the Office Action relies on a passage from *Fernandez* (col. 6, line 61 through col. 7, line 19) which discusses that a composition of view query (e.g., RXL view query) and a user query (e.g., XML-QL user query) can be used to produce a new view query from which an XML document results. The cited passage says nothing about the XML document being produced by one execution unit and consumed by another execution unit in performing the same execution plan, and says nothing about the form of the XML document being unusable by the consumer.

Next, for the Claim 1 limitation about the annotation that causes the XML data generated by the producer to be transformed to a canonical form for use by the consumer, the Office Action relies on a passage from *Fernandez* (col. 28, lines 1-5) which discusses a “compose” function that decomposes blocks of an XML-QL user query and “rewrites each nested pattern in a canonical form as a list of unnested patterns.” Applicants concede that *Fernandez* discusses transformation of XML-like data (a pattern) into a so-called canonical form. However, **the canonical form of *Fernandez* is simply a decomposition of a nested pattern into unnested patterns.** This says nothing about the multi-execution unit context of the subject matter of Claim 1 and nothing about the purpose of the transformation being to **transform XML data from a form that is unusable to a particular consumer into a form**

that is usable to the consumer, where both the producer and the consumer are working on portions of the same query execution plan.

Collectively, the foregoing distinctions between Claim 1 and the *Fernandez* reference illustrate how the disclosure of *Fernandez* is not a fair teaching of the subject matter recited in Claim 1. Hence, the disclosure of *Fernandez* cannot be stretched to form a valid anticipation rejection of Claim 1 and, therefore, Claim 1 is patentable thereover. **Claim 50** recites subject matter that is similar to the subject matter of Claim 1 and, therefore, Claim 50 is patentable over the cited reference of record for at least the same reasons as Claim 1.

Dependent **Claims 2-38** depend from Claim 1 and, therefore, are patentable over the cited reference of record for at least the same reasons as Claim 1. Furthermore, each of these dependent claims includes at least one other limitation that makes it further patentable over the reference of record. However, due to the fundamental differences between Claim 1 and *Fernandez* discussed above, discussion of these additional differences is unnecessary and is foregone at this time. The rejection of the dependent claims is collectively traversed, and no statements of official notice, overarching allegations of anticipation, or allegations of well-known features that may be present in the Office Action are stipulated to or admitted as prior art features, and the right to separately argue such features in the future is not disclaimed.

The distinctions between Claim 1 and *Fernandez* that are presented above are, generally, also applicable to the subject matter of independent **Claim 39** and **Claims 45**. That is, due to the deficiencies in the teachings of *Fernandez* regarding (a) a true query execution plan, (b) a multi-execution unit environment, and (c) transforming XML data generated by a particular producer execution unit from a form that is unusable to a particular consumer execution unit into a form that is usable to the consumer, *Fernandez* does not support a valid anticipation rejection of these claims.

Dependent **Claims 40-44** depend from Claim 39 and dependent **Claims 46-49** depend from Claim 45. Therefore, each of these dependent claims is patentable over the cited reference of record for at least the same reasons as the claim from which it depends. Furthermore, each of these dependent claims includes at least one other limitation that makes it further patentable over the reference of record. However, due to the fundamental differences between Claim 1 and *Fernandez* discussed above, discussion of these additional differences is unnecessary and is foregone at this time. The rejection of the dependent claims is collectively traversed, and no statements of official notice, overarching allegations of anticipation, or allegations of well-known features that may be present in the Office Action are stipulated to or admitted as prior art features, and the right to separately argue such features in the future is not disclaimed.

Based on the foregoing remarks, reconsideration and withdrawal of the rejection of Claims 1-50 under 35 U.S.C. § 102(e) is requested.

CONCLUSION

For the reasons set forth above, it is respectfully submitted that all of the pending claims are now in condition for allowance. Therefore, the issuance of a formal Notice of Allowance is believed next in order, and that action is most earnestly solicited.

The Examiner is respectfully requested to contact the undersigned by telephone if it is believed that such contact would further the examination of the present application.

Please charge any shortages or credit any overages to Deposit Account No. 50-1302.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP

Date: 3/5/07

John D. Henkhaus
John D. Henkhaus
Reg. No. 42,656

2055 Gateway Place, Suite 550
San Jose, CA 95110-1089
(408) 414-1080
Facsimile: (408) 414-1076

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: Mail Stop Amendment, Commissioner for Patents, P. O. Box 1450, Alexandria, VA 22313-1450

on 3/5/07 by Martina Placid
Martina Placid